Sub A6

We claim:

1. A computer system capable of accessing and controlling use of a watermarked software object, the system comprising:

a processor; and

a memory having computer executable instructions stored therein; and

wherein the processor, in response to the stored executable instructions:

reads a specific one of a plurality of watermarks embedded in the software object so as to yield an actual watermark value, wherein the specific one watermark is defined by a predefined value of a watermark key previously provided to and stored within the system; and

sets usage rights applicable to the object in response to the actual watermark value so as to control further use of the object by the computer system.

2. The system in claim 1 wherein the object is either a passive or active object, the passive object comprising content and the active object comprising executable code.

3. The system in claim 2 wherein, the processor, in response to the stored instructions and as part of the usage rights setting operation, supplies the usage rights to an operating system executing in the computer system in order to set a protection state applicable to the object.

4. The system in claim 3 wherein the watermark key expires after a predefined period of time elapses and the processor, in response to the stored instructions, obtains a new watermark key for subsequent use in lieu of the expired watermark key, wherein the new watermark key defines a different one of the plurality of watermarks embedded in the object.

5. The system in claim 3 wherein the value of the watermark key defines a pointer to a location in the object at which the specific one watermark appears.

6. The system in claim 5 wherein the location is a starting location.

7. The system in claim 5 wherein all of the plurality of said watermarks embedded in the object contain an identical watermark value.

8. The system in claim 7 wherein the identical watermark value contains a concatenation of a product identification value associated with the object and an identification value associated with the object provider.

9. The system in claim 3 wherein the processor, in response to the stored instructions:

reads a license for the object, the license specifying an expected value of a first parameter and the usage rights of the object;

compares the expected value of the first parameter against an actual value of the first parameter contained in the specific one watermark;

if the actual and expected values for first parameter do not identically match each other, prevents the object from being used.

10. The system in claim 9 wherein the processor, in response to the stored instructions:

obtains an expected value of a second parameter communicated with the specific one object;

extracts, from the specific one watermark detected in the object, the actual value of the first parameter and an actual value of the second parameter;

compares the expected values of the first and second parameters against the actual values of the first and second parameters, respectively; and

if the actual values of the first and second parameters identically and respectively match the expected values of the first and second parameters, permits the object to be used in accordance with the usage rights specified in the license.

11. The system in claim 10 wherein the processor, in response to the stored instructions, verifies that the license is signed by the object provider specified through the actual value of the second parameter found in the specific one watermark.

12. The system in claim 10 wherein the first and second parameters comprise a product identification (PID) value and a vendor identification (VID) value, respectively.

13. The system in claim 9 wherein the license a decryption key.

14. The system in claim 13 wherein the processor, in response to the stored instructions, generates a request for the license, wherein the request specifies the object.

15. The system in claim 14 wherein the request for the license further comprises a public key value associated with the computer system; and the license further comprises the expected value of the first parameter and the usage rights.

16. The system in claim 15 wherein the license further comprises a signature generated through use of a public key associated with a provider of the object, the signature being a function of the expected value of the first parameter and the usage rights.

17. The system in claim 16 wherein the processor, in response to the stored instructions:

performs a license verifying operation by:

verifying, using a predefined cryptographic parameter stored in the computer system, the public key associated with the object provider so as to define a certified public key of the object provider; and

verifying, using the certified public key of the object provider, the signature in the license as generated by the object provider so as to define a verified signature; and

performs an extraction operation by extracting, from the verified signature, the expected value of the first parameter, the encryption key and the usage rights.

18. The system in claim 17 wherein the value of the watermark key defines a pointer to a location in the object at which the specific one watermark appears.

19. The system in claim 18 wherein the location is a starting location.

20. The system in claim 18 wherein all of the plurality of said watermarks embedded in the object contain an identical watermark value.

21. The system in claim 20 wherein the identical watermark value contains a concatenation of a product identification value associated with the object, as the first parameter, and a vendor identification value associated with the object provider.

22. The system in claim 17 wherein the processor, in response to the stored instructions:

decrypts the object, as downloaded by the object provider to the computer system, using the decryption key specified in the license so as to yield a decrypted version of the object; and

reads the value of the specific one watermark in the decrypted version of the object.

23. The system in claim 22 wherein the decryption key is a symmetric encryption key which has been previously used, by the object provider, to encrypt the object in order to produce the encrypted version of the object.

24. The system in claim 22 wherein the watermark key expires after a predefined period of time elapses and the processor, in response to the stored instructions, obtains a new watermark key for subsequent use in lieu of the expired watermark key, wherein the new watermark key defines a different one of the plurality of watermarks embedded in the object.

25. The system in claim 22 further comprising an enforcer having:

an encrypted store for storing the encrypted version of the object produced by the object provider;

a decrypter for decrypting, using the decryption key, the encrypted version of the object stored in the encrypted store so as to yield a decrypted version of the object;

an unencrypted buffer for storing the decrypted object;

a watermark detector for detecting the presence of the specific one watermark embedded in the decrypted version of the object and for obtaining therefrom the actual value of the first parameter; and

a license verifier which:

performs the license verifying operation and, once the signature in the license is verified, the extraction operation so as to yield the decryption key, the expected value of the first parameter and the usage rights;

compares the expected value against the actual value of the first parameter; and

if the actual and expected values for first parameter do not identically match each other, then sets, in conjunction with the operating system, the protection state to prevent further use of the decrypted version of the object while the decrypted version remains in the unencrypted buffer.

26. The system in claim 25 wherein the processor, in response to the stored instructions:

obtains an expected value of a second parameter communicated with the specific one object;

extracts, from the specific one watermark detected in the object, the actual value of the first parameter and an actual value of the second parameter;

compares the expected values of the first and second parameters against the actual values of the first and second parameters, respectively; and

if the actual values of the first and second parameters identically and respectively match the expected values of the first and second parameter sets, in conjunction with the operating system and consistent with the usage rights, the protection state to govern use of the decrypted version of the object while the decrypted version remains in the unencrypted buffer.

27. The system in claim 26 wherein the first and second parameters comprise a product identification (PID) value and a vendor identification (VID) value, respectively.

28. The system in claim 25 wherein if the license exists for the object, the processor, in response to the stored instructions and through the license verifier, sets the usage rights to appropriate values so as to inhibit further use of the decrypted object if the watermark detector fails to detect the specific one watermark in the decrypted version of the object.

29. The system in claim 28 wherein either all or a portion of the enforcer is located either in the operating system or in a media card associated with the computer system.

30. The system in claim 28 wherein the operating system comprises a digital rights management system having a license database which stores the license, and, subsequently, in response to a request issued by the computer system to access the object, provides the license to the enforcer.

31. The system in claim 30 wherein the request for the license further comprises an authorization for payment of a predefined fee in exchange for the license.

32. The system in claim 28 wherein the value of the watermark key defines a pointer to a location in the object at which the specific one watermark appears.

33. The system in claim 32 wherein the location is a starting location.

34. The system in claim 32 wherein all of the plurality of said watermarks embedded in the object contain an identical watermark value.

35. The system in claim 34 wherein the identical watermark value contains a concatenation of a product identification value associated with the object, as the first parameter, and a vendor identification value associated with the object provider.

36. The system in claim 28 wherein the decryption key is a symmetric encryption key which has been previously used, by the object provider, to encrypt the object in order to produce the encrypted version of the object.

37. The system in claim 28 wherein the watermark key expires after a predefined period of time elapses and the processor, in response to the stored instructions, obtains a new watermark key for subsequent use in lieu of the expired watermark key, wherein the new watermark key defines a different one of the plurality of watermarks embedded in the object.

38. The system in claim 3 wherein the processor, in response to the stored instructions, downloads the object, via a network connection, from a first server.

39. The system in claim 38 wherein the watermark key expires after a predefined period of time elapses and the processor, in response to the stored instructions, obtains a new watermark key for subsequent use in lieu of the expired watermark key, wherein the new watermark key defines a different one of the plurality of watermarks embedded in the object.

40. The system in claim 38 wherein the value of the watermark key defines a pointer to a location in the object at which the specific one watermark appears.

41. The system in claim 40 wherein the location is a starting location.

42. The system in claim 40 wherein all of the plurality of said watermarks embedded in the object contain an identical watermark value.

43. The system in claim 42 wherein the identical watermark value contains a concatenation of a product identification value associated with the object and a vendor identification value associated with the object provider.

44. The system in claim 38 wherein the processor, in response to the stored instructions:

reads a license for the object, the license specifying an expected value of a first parameter and the usage rights of the object;

compares the expected value of the first parameter against an actual value of the first parameter contained in the specific one watermark;

if the actual and expected values for first parameter do not identically match each other, prevents the object from being used.

45. The system in claim 44 wherein the processor, in response to the stored instructions:

obtains an expected value of a second parameter communicated with the specific one object;

extracts, from the specific one watermark detected in the object, the actual value of the first parameter and an actual value of the second parameter;

compares the expected values of the first and second parameters against the actual values of the first and second parameters, respectively; and

if the actual values of the first and second parameters identically and respectively match the expected values of the first and second parameters, permits the object to be used in accordance with the usage rights specified in the license.

46. The system in claim 45 wherein the processor, in response to the stored instructions, verifies that the license is signed by the object provider specified through the actual value of the second parameter found in the specific one watermark.

47. The system in claim 45 wherein the first and second parameters comprise a product identification (PID) value and a vendor identification (VID) value, respectively.

48. The system in claim 44 wherein the license comprises a decryption key.

49. The system in claim 40 wherein the processor, in response to the stored instructions, obtains the license from a second server and via a network connection existing between the computer system and the second server.

50. The system in claim 49 wherein the first and second servers are the same.

51. The system in claim 49 wherein the request for the license further comprises a public key value associated with the computer system; and the license further comprises the expected value of the first parameter and the usage rights.

52. The system in claim 51 wherein the processor, in response to the stored instructions, generates a request, via a network connection, to the second server for the license, wherein the request specifies the object.

53. The system in claim 52 wherein the license further comprises a signature generated through use of a public key associated with a provider of the object, the

signature being a function of the expected value of the first parameter and the usage rights.

54. The system in claim 53 wherein the processor, in response to the stored instructions:

performs a license verifying operation by:

verifying, using a predefined cryptographic parameter stored in the computer system, the public key associated with the object provider so as to define a certified public key of the object provider; and

verifying, using the certified public key of the object provider, the signature in the license as generated by the object provider so as to define a verified signature; and

performs an extraction operation by extracting, from the verified signature, the expected value of the first parameter, the encryption key and the usage rights.

55. The system in claim 54 wherein the value of the watermark key defines a pointer to location in the object at which the specific one watermark appears.

56. The system in claim 55 wherein the location is a starting location.

57. The system in claim 55 wherein all of the plurality of said watermarks embedded in the object contain an identical watermark value.

58. The system in claim 57 wherein the identical watermark value contains a concatenation of a product

identification value associated with the object, as the first parameter, and a vendor identification value associated with the object provider.

59. The system in claim 54 wherein the processor, in response to the stored instructions:

decrypts the object, as downloaded by the object provider to the computer system, using the decryption key specified in the license so as to yield a decrypted version of the object; and

reads the value of the specific one watermark in the decrypted version of the object.

60. The system in claim 59 wherein the decryption key is a symmetric encryption key which has been previously used, by the object provider, to encrypt the object in order to produce the encrypted version of the object.

61. The system in claim 59 wherein the watermark key expires after a predefined period of time elapses and the processor, in response to the stored instructions, obtains a new watermark key for subsequent use in lieu of the expired watermark key, wherein the new watermark key defines a different one of the plurality of watermarks embedded in the object.

62. The system in claim 59 further comprising an enforcer having:

an encrypted store for storing the encrypted version of the object produced by the object provider;

a decrypter for decrypting, using the decryption key, the encrypted version of the object stored in the encrypted store so as to yield a decrypted version of the object;

an unencrypted buffer for storing the decrypted object;

a watermark detector for detecting the presence of the specific one watermark embedded in the decrypted version of the object and for obtaining therefrom the actual value of the first parameter; and

a license verifier which:

performs the license verifying operation and, once the signature in the license is verified, the extraction operation so as to yield the decryption key, the expected value of the first parameter and the usage rights;

compares the expected value against the actual value of the first watermark; and

if the actual and expected values for first parameter do not identically match each other, then sets, in conjunction with the operating system, the protection state to prevent further use of the decrypted version of the object while the decrypted version remains in the unencrypted buffer.

63. The system in claim 62 wherein the processor, in response to the stored instructions:

obtains an expected value of a second parameter communicated with the specific one object;

extracts, from the specific one watermark detected in the object, the actual value of the first parameter and an actual value of the second parameter;

compares the expected values of the first and second parameters against the actual values of the first and second parameters, respectively; and

if the actual value of the first and second parameters identically and respectively match the expected values of the first and second parameter sets, in conjunction with the operating system and consistent with the usage rights, the protection state to govern use of the decrypted version of the object while the decrypted version remains in the unencrypted buffer.

64. The system in claim 63 wherein the first and second parameters comprise a product identification (PID) value and a vendor identification (VID) value, respectively.

65. The system in claim 62 wherein if the license exists for the object, the processor, in response to the stored instructions and through the license verifier, sets the usage rights to appropriate values so as to inhibit further use of the decrypted object if the watermark detector fails to detect the specific one watermark in the decrypted version of the object.

66. The system in claim 65 wherein either all or a portion of the enforcer is located either in the operating system or in a media card associated with the computer system.

67. The system in claim 65 wherein the operating system comprises a digital rights management system having a license database which stores the license, and, subsequently, in response to a request issued by the computer system to access the object, provides the license to the enforcer.

68. The system in claim 67 wherein the request for the license further comprises an authorization for payment of a predefined fee in exchange for the license.

69. The system in claim 65 wherein the value of the watermark key defines a pointer to a location in the object at which the specific one watermark appears.

70. The system in claim 69 wherein the location is a starting location.

71. The system in claim 69 wherein all of the plurality of said watermarks embedded in the object contain an identical watermark value.

72. The system in claim 71 wherein the identical watermark value contains a concatenation of a product identifier associated with the object and an identifier associated with the object provider.

73. The system in claim 62 wherein the first and second servers are the same.

74. The system in claim 62 wherein the decryption key is a symmetric encryption key which has been previously used, by the object provider, to encrypt the object in order to produce the encrypted version of the object.

75. The system in claim 62 wherein the watermark key expires after a predefined period of time elapses and the processor, in response to the stored instructions, obtains a new watermark key for subsequent use in lieu of the expired watermark key, wherein the new watermark key defines a different one of the plurality of watermarks embedded in the object.

76. The system in claim 62 wherein the processor, in response to the stored instructions, obtains the new watermark key, via a network connection, from a third server.

77. The system in claim 62 wherein the third server is either the same as the first or second server, or is associated with a third party watermarking authority.

78. The system in claim 77 wherein the first and second servers are the same.

79. In a computer system having a processor and a memory having computer executable instructions stored therein, a method, implemented through execution of the stored instructions, for accessing and controlling use of a watermarked software object comprising the steps of:

reading a specific one of a plurality of watermarks embedded in the software object so as to yield an actual watermark value, wherein the specific one watermark is defined by a predefined value of a watermark key previously provided to and stored within the system; and

setting usage rights applicable to the object in response to the actual watermark value so as to control further use of the object by the computer system.

80. The method in claim 79 wherein the object is either a passive or active object, the passive object comprising content and the active object comprising executable code.

81. The method in claim 80 wherein the usage rights setting step comprises the step of supplying the usage rights to an operating system executing in the computer system in order to set a protection state applicable to the object.

82. The method in claim 81, wherein the watermark key expires after a predefined period of time elapses, further comprising the step of obtaining a new watermark key for subsequent use in lieu of the expired watermark key, wherein the new watermark key defines a different one of the plurality of watermarks embedded in the object.

83. The method in claim 81 wherein the value of the watermark key defines a pointer to a location in the object at which the specific one watermark appears.

84. The method in claim 83 wherein the location is a starting location.

85. The method in claim 83 wherein all of the plurality said watermarks embedded in the object contain an identical watermark value.

86. The method in claim 85 wherein the identical watermark value contains a concatenation of a product identification value associated with the object and an identification value associated with the object provider.

87. The method in claim 81 comprising the steps of:

reading a license for the object, the license specifying an expected value of a first parameter and the usage rights of the object:

comparing the expected value of the first parameter against an actual value of the first parameter contained in the specific one watermark;

if the actual and expected values for first parameter do not identically match each other, preventing the object from being used.

88. The method in claim 87 comprising the steps of:

obtaining an expected value of a second parameter communicated with the specific one object;

extracting, from the specific one watermark detected in the object, the actual value of the first parameter and an actual value of the second parameter;

comparing the expected values of the first and second parameters against the actual values of the first and second parameters, respectively; and

if the actual values of the first and second parameters identically and respectively match the expected values of the first and second parameters, permitting the object to be used in accordance with the usage rights specified in the license.

89. The method in claim 88 comprising the step of verifying that the license is signed by the object provider specified through the actual value of the second parameter found in the specific one watermark.

90. The method in claim 88 wherein the first and second parameters comprise a product identification (PID) value and a vendor identification (VID) value, respectively.

91. The method in claim 87 wherein the license comprises a decryption key.

92. The method in claim 91 comprising the step of generating a request for the license, wherein the request specifies the object.

93. The method in claim 92 wherein the request for the license further comprises a public key value associated with the computer system; and the license further comprises the expected value of the first parameter and the usage rights.

94. The method in claim 93 wherein the license further comprises a signature generated through use of a public key associated with a provider of the object, the signature being a function of the expected watermark value, the usage rights.

95. The method in claim 94 further comprising the steps of:

performing a license verifying operation by:

verifying, using a predefined cryptographic parameter stored in the computer system, the public key associated with the object provider so as to define a certified public key of the object provider; and

verifying, using the certified public key of the object provider, the signature in the license as generated by the object provider so as to define a verified signature; and

performing an extraction operation by extracting, from the verified signature, the expected value of the first parameter, the encryption key and the usage rights.

96. The method in claim 95 wherein the value of the watermark key defines a pointer to a location in the object at which the specific one watermark appears.

97. The method in claim 96 wherein the location is a starting location.

98. The method in claim 96 wherein all of the plurality of said watermarks embedded in the object contain an identical watermark value.

99. The method in claim 98 wherein the identical watermark value contains a concatenation of a product identification value associated with the object, as the first parameter, and a vendor identification value associated with the object provider.

100. The method in claim 95 comprising the steps of:

decrypting the object, as downloaded by the object provider to the computer system, using the decryption key specified in the license so as to yield a decrypted version of the object; and

reading the value of the specific one watermark in the decrypted version of the object.

101. The method in claim 100 wherein the decryption key is a symmetric encryption key which has been previously used, by the object provider, to encrypt the object in order to produce the encrypted version of the object.

102. The method in claim 100, wherein the watermark key expires after a predefined period of time elapses, further comprising the step of obtaining a new watermark key for subsequent use in lieu of the expired watermark key, wherein the new watermark key defines a different one of the plurality of watermarks embedded in the object.

103. The method in claim 81 further comprising the step of downloading the object, via a network connection, from a first server.

104. The method in claim 103, wherein the watermark key expires after a predefined period of time elapses, further comprising the step of obtaining a new watermark key for subsequent use in lieu of the expired watermark key, wherein the new watermark key defines a different one of the plurality of watermarks embedded in the object.

105. The method in claim 103 wherein the value of the watermark key defines a pointer to a location in the object at which the specific one watermark appears.

106. The method in claim 105 wherein the location is a starting location.

107. The method in claim 105 wherein all of the plurality of said watermarks embedded in the object contain an identical watermark value.

108. The method in claim 107 wherein the identical watermark value contains a concatenation of a product identification value associated with the object and a vendor identification value associated with the object provider.

109. The method in claim 103 comprising the steps of:

reading a license for the object, the license specifying an expected value of a first parameter and the usage rights of the object:

comparing the expected value of the first parameter against the actual value of the first parameter contained in the specific one watermark;

if the actual and expected values for first parameter do not identically match each other, preventing the object from being used.

110. The method in claim 109 further comprising the steps of:

obtaining an expected value of a second parameter communicated with the specific one object;

extracting, from the specific one watermark detected in the object, the actual value of the first parameter and an actual value of the second parameter;

comparing the expected values of the first and second parameters against the actual values of the first and second parameters, respectively; and

if the actual values of the first and second parameters identically and respectively match the expected values of the first and second parameters, permitting the object to be used in accordance with the usage rights specified in the license.

111. The method in claim 110 further comprising the step of verifying that the license is signed by the object provider specified through the actual value of the second parameter found in the specific one watermark.

112. The method in claim 110 wherein the first and second parameters comprise a product identification (PID) value and a vendor identification (VID) value, respectively.

113. The method in claim 109 wherein the license comprises a decryption key.

114. The method in claim 105 further comprising the step of obtaining the license from a second server and via a network connection existing between the computer system and the second server.

115. The method in claim 114 wherein the request for the license further comprises a public key value associated with the computer system; and the license further comprises the expected value of the first parameter and the usage rights.

116. The method in claim 115 further comprising the step of generating a request, via a network connection, to the second server for the license, wherein the request specifies the object.

117. The method in claim 116 wherein the license further comprises a signature generated through use of a public key associated with a provider of the object, the signature being a function of the expected watermark value, the usage rights.

118. The method in claim 116 further comprising the steps of:

performing a license verifying operation by:

verifying, using a predefined cryptographic parameter stored in the computer system, the public key

associated with the object provider so as to define a certified public key of the object provider; and

verifying, using the certified public key of the object provider, the signature in the license as generated by the object provider so as to define a verified signature; and

performing an extraction operation by extracting, from the verified signature, the expected value of the first parameter, the encryption key and the usage rights.

119. The method in claim 118 wherein the value of the watermark key defines a pointer to a location in the object at which the specific one watermark appears.

120. The method in claim 119 wherein the location is a starting location.

121. The method in claim 119 wherein all of the plurality of said watermarks embedded in the object contain an identical watermark value.

122. The method in claim 121 wherein the identical watermark value contains a concatenation of a product identification value associated with the object, as the first parameter, and a vendor identification value associated with the object provider.

123. The method in claim 118 further comprising the steps of:

decrypting the object, as downloaded by the object provider to the computer system, using the decryption key

specified in the license so as to yield a decrypted version of the object; and

reading the value of the specific one watermark in the decrypted version of the object.

124. The method in claim 59 wherein the decryption key is a symmetric encryption key which has been previously used, by the object provider, to encrypt the object in order to produce the encrypted version of the object.

125. The method in claim 59, wherein the watermark key expires after a predefined period of time elapses, further comprising the step of obtaining a new watermark key for subsequent use in lieu of the expired watermark key, wherein the new watermark key defines a different one of the plurality of watermarks embedded in the object.

126. A computer readable medium having computer executable instructions stored therein for performing the steps of claim 79.

127. Apparatus for a networked client-server environment, for accessing a software object from a first server and using the object so accessed, the apparatus comprising:

a client computer connected to the network, the client computer having:

a processor; and

a memory having computer executable instructions stored therein; and

wherein the processor, in response to the stored executable instructions:

issues, in response to input information, a download request to the first server to download a file containing a software object;

obtains the file containing a watermarked version of the software object from the first server;

reads a specific one of a plurality of watermarks embedded in the software object downloaded from the first server so as to yield an actual watermark value, wherein the specific one watermark is defined by a predefined value of a watermark key previously provided to and stored within the client computer; and

sets usage rights applicable to the object in response to the actual watermark value so as to control further use of the object by the client computer; and

the first server connected to the network, wherein the server:

in response to the download request, accesses the watermarked version of the software object, wherein a plurality of watermarks have been embedded into the object, and downloading the file containing the watermarked version of the software object to the client computer.

128. The apparatus in claim 127 wherein the software object is either a passive or active object, the passive object comprising content and the active object comprising executable code.

129. The apparatus in claim 128 wherein, the processor, in response to the stored instructions and as part of the usage rights setting operation, supplies the usage rights to an operating system executing in the client computer in order to set a protection state applicable to the software object.

130. The apparatus in claim 129 wherein the value of the watermark key defines a pointer to a location in the software object at which the specific one watermark appears.

131. The apparatus in claim 130 wherein the location is a starting location.

132. The apparatus in claim 130 wherein all of the plurality of said watermarks embedded in the software object contain an identical watermark value.

133. The apparatus in claim 130 wherein

the processor:

issues, in response to further input information, a request to a second server to obtain a license to use the software object, wherein the request specifies the software object;

compares an expected value of a first parameter contained in the license against an actual value of the first parameter contained in the specific one watermark;

if the actual and expected values for the first parameter do not identically match each other,

prevents the software object from being used by the client computer; and

the first server, in response to the license request:

generates a license specifying the expected value of the first parameter and the usage rights of the software object accorded to the client computer by the object provider; and

transmits the license, via the network, to the client computer.

134. The apparatus in claim 133 wherein the processor, in response to the stored instructions:

obtains an expected value of a second parameter communicated with the specific one object;

extracts, from the specific one watermark detected in the object, the actual value of the first parameter and an actual value of the second parameter;

compares the expected values of the first and second parameters against the actual values of the first and second parameters, respectively; and

if the actual values of the first and second parameters identically and respectively match the expected values of the first and second parameters, permits the object to be used in accordance with the usage rights specified in the license.

135. The apparatus in claim 134 wherein the processor in response to the stored instructions, verifies that the license is signed by the object provider specified

through the actual value of the second parameter found in the specific one watermark.

136. The apparatus in claim 134 wherein the first and second parameters comprise a product identification (PID) value and a vendor identification (VID) value, respectively.

137. The apparatus in claim 133 wherein the license further comprises a decryption key.

138. The apparatus in claim 137 wherein the request for the license further comprises a public key value associated with a provider of the object and a computer identification value both associated with the client computer.

139. The apparatus in claim 138 wherein the server, in response to the license request:

accesses the watermarked object specified in the request;

encrypts the watermarked object using a predefined encryption key; and

generates a cryptographic signature using a public key associated with the provider of the object, wherein the signature is a function of the expected value of the first parameter and the usage rights.

140. The apparatus in claim 139 wherein the license request further comprises a computer identification number associated with the client computer, and the file downloaded to the client computer further comprises the public key of the server.

141. The apparatus in claim 140 wherein the server:

establishes, in response to the request, an entry in a database associating the particular copy of the software object with the encryption key; and

subsequently, in conjunction with issuing the license and in response to the computer identification value of the client computer, updates the entry to associate the particular copy of the software object with client computer.

142. The apparatus in claim 141 wherein the server, prior to encrypting the object, provides a fingerprint value with the object, the fingerprint uniquely identifying a particular copy of the object to be downloaded to the client computer, so as to define a fingerprinted watermarked object which, in turn, is downloaded to the client computer as the watermarked version of the software object.

143. The apparatus in claim 140 wherein the processor, in response to the stored instructions:

performs a license verifying operation by:

verifying, using a predefined cryptographic parameter stored in the client computer, the public key

associated with the object provider so as to define a certified public key of the object provider; and

verifying, using the certified public key of the object provider, the signature in the license as generated by the object provider so as to define a verified signature; and

performs an extraction operation by extracting, from the verified signature, the expected value of the first parameter, the encryption key and the usage rights.

144. The apparatus in claim 143 wherein the value of the watermark key defines a pointer to a location in the watermarked object at which the specific one watermark appears.

145. The apparatus in claim 144 wherein the location is a starting location.

146. The apparatus in claim 143 wherein all of the plurality of said watermarks embedded in the object contain an identical watermark value.

147. The apparatus in claim 146 wherein the identical watermark value contains a concatenation of a product identification value associated with the object, as the first parameter, and a vendor identification value associated with the object provider.

148. The apparatus in claim 146 wherein the processor, in response to the stored instructions:

decrypts the object, as downloaded by the object provider to the client computer, using the decryption key specified in the license so as to yield a decrypted version of the object; and

reads the value of the specific one watermark in the decrypted version of the object.

149. The apparatus in claim 148 wherein the decryption key is a symmetric encryption key which has been previously used, by the object provider, to encrypt the object in order to produce the encrypted version of the object.

150. The apparatus in claim 143 wherein the computer identification value is a processor serial number.

151. The apparatus in claim 143 wherein the first and second servers are the same.

152. The apparatus in claim 143 wherein the watermark values contains a concatenation of a product identification value associated with the software object, as the first parameter, and a vendor identification value associated with the object provider.

153. In a networked client-server environment, a method for accessing a software object from a first server and using the object so accessed, the method comprising the steps of:

in a client computer connected to the network, the client computer having a processor, and a memory having computer executable instructions stored therein, the steps, performed in response to the executable instructions, of and

issuing, in response to input information, a download request to the first server to download a file containing a software object;

obtaining the file containing a watermarked version of the software object from the first server;

reading a specific one of a plurality of watermarks embedded in the software object downloaded from the first server so as to yield an actual watermark value, wherein the specific one watermark is defined by a predefined value of a watermark key previously provided to and stored within the client computer; and

setting usage rights applicable to the object in response to the actual watermark value so as to control further use of the object by the client computer; and

in the first server connected to the network, the steps, in response to the download request of:

accessing the watermarked version of the software object, wherein a plurality of watermarks have been embedded into the object; and

downloading the file containing the watermarked version of the software object to the client computer.

154. The method in claim 153 wherein the software object is either a passive or active object, the passive object comprising content and the active object comprising executable code.

155. The method in claim 154 wherein the usage rights setting step comprises the step of supplying the usage rights to an operating system executing in the client computer in order to set a protection state applicable to the software object.

156. The method in claim 155 wherein the value of the watermark key defines a pointer to a location in the software object at which the specific one watermark appears.

157. The method in claim 156 wherein the location is a starting location.

158. The method in claim 156 wherein all of the plurality of said watermarks embedded in the software object contain an identical watermark value.

159. The method in claim 156 further comprising the steps of:

in the client computer:

issuing, in response to further input information, a request to a second server to obtain a license to use the software object, wherein the request specifies the software object;

comparing an expected value of a first parameter contained in the license against an actual value of the first parameter contained in the specific one watermark; and

if the actual and expected values for first parameter do not identically match each other, preventing the software object from being used by the client computer; and

in the first server, in response to the license request:

generating a license specifying the expected value of the first parameter and the usage rights of the software object accorded to the client computer by the object provider; and

transmitting the license, via the network, to the client computer.

160. The method in claim 159 further comprising the steps, in the client computer, of:

obtaining an expected value of a second parameter communicated with the specific one object;

extracting, from the specific one watermark detected in the object, the actual value of the first parameter and an actual value of the second parameter;

comparing the expected values of the first and second parameters against the actual values of the first and second parameters, respectively; and

if the actual values of the first and second parameters identically and respectively match the expected values of the first and second parameters,

permitting the object to be used in accordance with the usage rights specified in the license.

161. The method in claim 160 further comprising the step, in the client computer, of verifying that the license is signed by the object provider specified through the actual value of the second parameter found in the specific one watermark.

162. The method in claim 160 wherein the first and second parameters comprise a product identification (PID) value and a vendor identification (VID) value, respectively.

163. The method in claim 159 wherein the license further comprises a decryption key.

164. The method in claim 163 wherein the request for the license further comprises a public key value associated with a provider of the object and a computer identification value both associated with the client computer.

165. The method in claim 164 further comprising the steps, in the server and in response to the license request, of:

accessing the watermarked object specified in the request;

encrypting the watermarked object using a predefined encryption key; and

generating a cryptographic signature using a public key associated with the provider of the object, wherein

the signature is a function of the expected value of the first parameter and the usage rights.

166. The method in claim 165 wherein the license request further comprises a computer identification number associated with the client computer, and the file downloaded to the client computer further comprises the public key of the server.

167. The method in claim 166 further comprising the steps, in the server, of:

establishing, in response to the request, an entry in a database associating the particular copy of the software object with the encryption key; and

subsequently, in conjunction with issuing the license and in response to the computer identification value of the client computer, updating the entry to associate the particular copy of the software object with client computer.

168. The method in claim 167 further comprising the steps, in the server and, prior to encrypting the object, of providing a fingerprint value with the object, the fingerprint uniquely identifying a particular copy of the object to be downloaded to the client computer, so as to define a fingerprinted watermarked object which, in turn, is downloaded to the client computer as the watermarked version of the software object.

169. The method in claim 166 further comprising the steps, in the client computer, of:

verifying, using a predefined cryptographic parameter stored in the client computer, the public key associated with the object provider so as to define a certified public key of the object provider; and

verifying, using the certified public key of the object provider, the signature in the license as generated by the object provider so as to define a verified signature; and

extracting, from the verified signature, the expected value of the first parameter, the encryption key and the usage rights.

170. The method in claim 169 wherein the value of the watermark key defines a pointer to a location in the watermarked object at which the specific one watermark appears.

171. The method in claim 170 wherein the location is a starting location.

172. The method in claim 169 wherein all of the plurality of said watermarks embedded in the object contain an identical watermark value.

173. The method in claim 172 wherein the identical watermark value contains a concatenation of a product identification value associated with the object, as the first parameter, and a vendor identification value associated with the object provider.

174. The method in claim 172 further comprising the steps, in the client computer, of:

decrypting the object, as downloaded by the object provider to the client computer, using the decryption key specified in the license so as to yield a decrypted version of the object; and

reading the value of the specific one watermark in the decrypted version of the object.

175. The method in claim 174 wherein the decryption key is a symmetric encryption key which has been previously used, by the object provider, to encrypt the object in order to produce the encrypted version of the object.

176. In a networked client-server environment, apparatus for use in conjunction with a digital rights management system, the apparatus comprising:

a client computer connected to the network, the client computer having:

a processor;

a memory having computer executable instructions stored therein; and

an enforcer, contained within the digital rights management system, for controlling use of watermarked software objects, wherein the enforcer stores a predefined watermark key which defines a specific one of a plurality of watermarks embedded in the watermarked software object to be used by the enforcer in subsequently controlling use of each one of said watermarked software objects, and wherein the watermark

key expires after a predefined period of time elapses since said key was initially stored in the enforcer;

wherein the processor, in response to the stored executable instructions:

establishes a network connection to a server;

issues a request to the server for a new watermark key; and

utilizes either the predefined watermark key or the new watermark key, as received from the server, for the predefined watermark key for subsequent use in controlling access to the watermarked software objects until such time as the predefined key has expired after which the new watermark key is used instead; and

the server, connected to the network, which, in response to the request:

selects, if the predefined key has not been revoked for the client computer, another one of a predefined plurality of predetermined watermark keys for use in controlling access to the software watermarks objects as the new watermark key;

sends the new watermark key to the client computer; and

if the predefined key has been revoked, does not supply the new watermark key to the client computer.

177. The apparatus in claim 176 wherein the network connection comprises a secure connection.

178. The apparatus in claim 177 wherein the server is associated with a publisher of any one of the watermarked software objects or a vendor of said one object, or a watermarking authority.

179. The apparatus in claim 178 wherein:

the client computer, in response to the stored instructions and in conjunction with the request, also supplies the server with an existing certificate for a predefined public key associated with the client computer; and

the server, if the existing certificate for the public key has not been revoked by the server, provides the client computer with the new watermark key.

180. In a networked client-server environment, a method for use in conjunction with a digital rights management system,

in a client computer connected to a network, the client computer having: a processor; a memory having computer executable instructions stored therein; and an enforcer, contained within the digital rights management system, for controlling use of watermarked software objects, wherein the enforcer stores a predefined watermark key which defines a specific one of a plurality of watermarks embedded in the watermarked software object to be used by the enforcer in subsequently controlling use of each one of said watermarked software objects, and wherein the watermark key expires after a predefined period of time elapses since said key was initially stored in the enforcer; wherein the method comprises the

steps, upon expiration of the watermark key, performed by the processor, in response to the stored executable instructions, of:

establishing a network connection to a server;

issuing a request to the server for a new watermark key; and

utilizes either the predefined watermark key or the new watermark key, as received from the server, for the predefined watermark key for subsequent use in controlling access to the watermarked software objects until such time as the predefined key has expired after which the new watermark key is used instead; and

in the server, connected to the network and, in response to the request, the steps of:

selecting, only if the predefined key has not been revoked for the client computer, another one of a predefined plurality of predetermined watermark keys for use in controlling access to the software watermarks objects as the new watermark key;

sending the new watermark key to the client computer; and

if the predefined key has been revoked, not sending the new watermark key to the client computer.

181. The method in claim 180 wherein the network connection comprises a secure connection.

182. The method in claim 181 wherein the server is associated with a publisher of any one of the watermarked software objects or a vendor of said one object, or a watermarking authority.

183. The method in claim 182 further comprising the steps of:

in the client computer and in response to the stored instructions and in conjunction with the request:

supplying the server with an existing certificate for a predefined public key associated with the client computer; and

in the server, if the existing certificate for the public key has not been revoked by the server, providing the client computer with a new certificate, for the new watermark key.

184. In a networked client-server environment, apparatus for obtaining a watermark key for use in a digital rights management system, the apparatus comprising:

a client computer connected to the network, the client computer having:

a processor;

a memory having computer executable instructions stored therein; and

an enforcer, contained within the digital rights management system, for controlling use of watermarked software objects, wherein the enforcer is capable of storing a predefined watermark key which defines a specific one of a plurality of watermarks embedded in the watermarked software object to be used by the enforcer in subsequently controlling use of each one of said watermarked software objects;

wherein, if the enforcer does not then possess the watermark key, the processor, in response to the stored executable instructions:

establishes a network connection to a server;

issues a request to the server for a watermark key; and

stores the watermark key, received from the server, within the enforcer for subsequent use in controlling access to watermarked software objects; and

the server, connected to the network, which, in response to the request:

selects, one of a predefined plurality of predetermined watermark keys for use in controlling access to the software watermarked objects as the watermark key;

downloads the watermark key to the client computer.

185. The apparatus in claim 184 wherein the request contains a public key associated with the client computer and

the server, in response to the request:

encrypts the watermark key using the public key of the client computer so as to yield the encrypted key; and

downloads the encrypted key to the client computer as the watermark key; and

the client computer:

upon receipt of the watermark key, decrypts the encrypted key using a private key associated with the client computer so as to yield a decrypted key; and

stores the decrypted key as the watermark key.

186. The apparatus in claim 185 wherein the network connection comprises a secure connection.

187. The apparatus in claim 186 wherein the server is associated with a publisher of any one of the watermarked software objects or a vendor of said one object, or a watermarking authority.

188. In a networked client-server environment, a method for obtaining a watermark key for use in a digital rights management system,

in a client computer connected to a network, the client computer having: a processor; a memory having computer executable instructions stored therein; and an enforcer, contained within the digital rights management system, for controlling use of watermarked software objects, wherein the enforcer is capable of storing a predefined watermark key which defines a specific one of a plurality of watermarks embedded in the watermarked software object to be used by the enforcer in subsequently controlling use of each one of said watermarked software objects; wherein the method comprises the steps, performed by the processor, if the enforcer does not then possess the watermark key and in response to the stored executable instructions, of:

establishing a network connection to a server;

issuing a request to the server for a watermark key; and

storing the watermark key, received from the server, within the enforcer for subsequent use in controlling access to watermarked software objects; and

in the server, connected to the network and in response to the request:

selecting, one of a predefined plurality of predetermined watermark keys for use in controlling access to the software watermarked objects as the watermark key;

downloading the watermark key to the client computer.

189. The method in claim 188, wherein the request contains a public key associated with the client computer, comprising the steps of:

in the server, in response to the request:

encrypting the watermark key using the public key of the client computer so as to yield the encrypted key; and

downloading the encrypted key to the client computer as the watermark key; and

in the processor, in response to the stored instructions:

upon receipt of the watermark key, decrypting the encrypted key using a private key associated with the client computer so as to yield a decrypted key; and

storing the decrypted key as the watermark key.

190. The method in claim 189 wherein the network connection comprises a secure connection.

191. The method in claim 190 wherein the server is associated with a publisher of any one of the watermarked

software objects or a vendor of said one object, or a watermarking authority.